



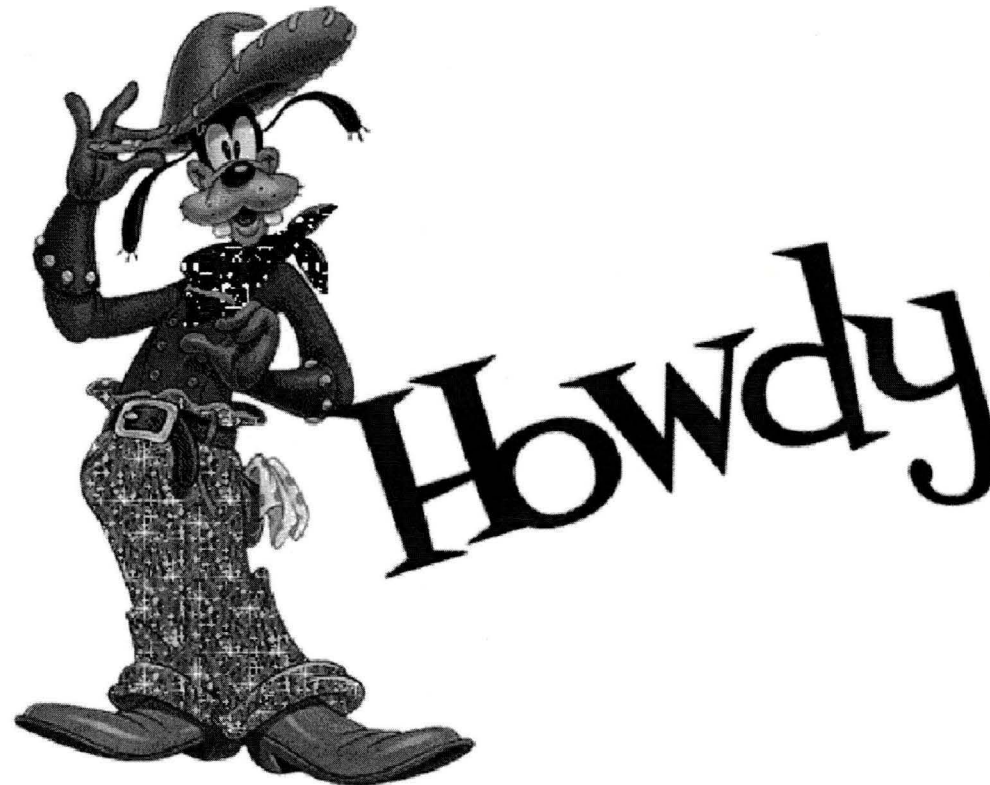
Watermarking — Paving the Way for Better Released Documents at KSC

A decorative graphic consisting of numerous overlapping circles and teardrop shapes of various sizes, arranged in a horizontal band across the lower half of the page.

Keith Little, Carole Miller, Paul Schwindt
June, 2012

- To talk about implementing Watermarks for ProE and Office Documents
 - KSC implementation is in 9.1, but applicable to 10.x as well

- Background
- the Problem
- the Premise
- Implementation
- Specifics
- Challenges



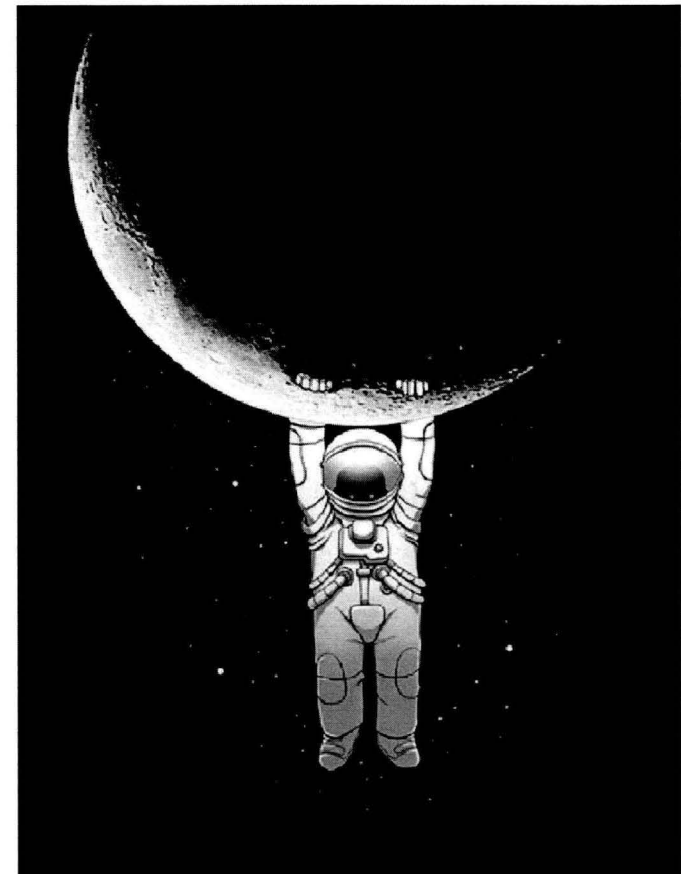
About Us



NASA Design Engineering

Kennedy Space Center, FL

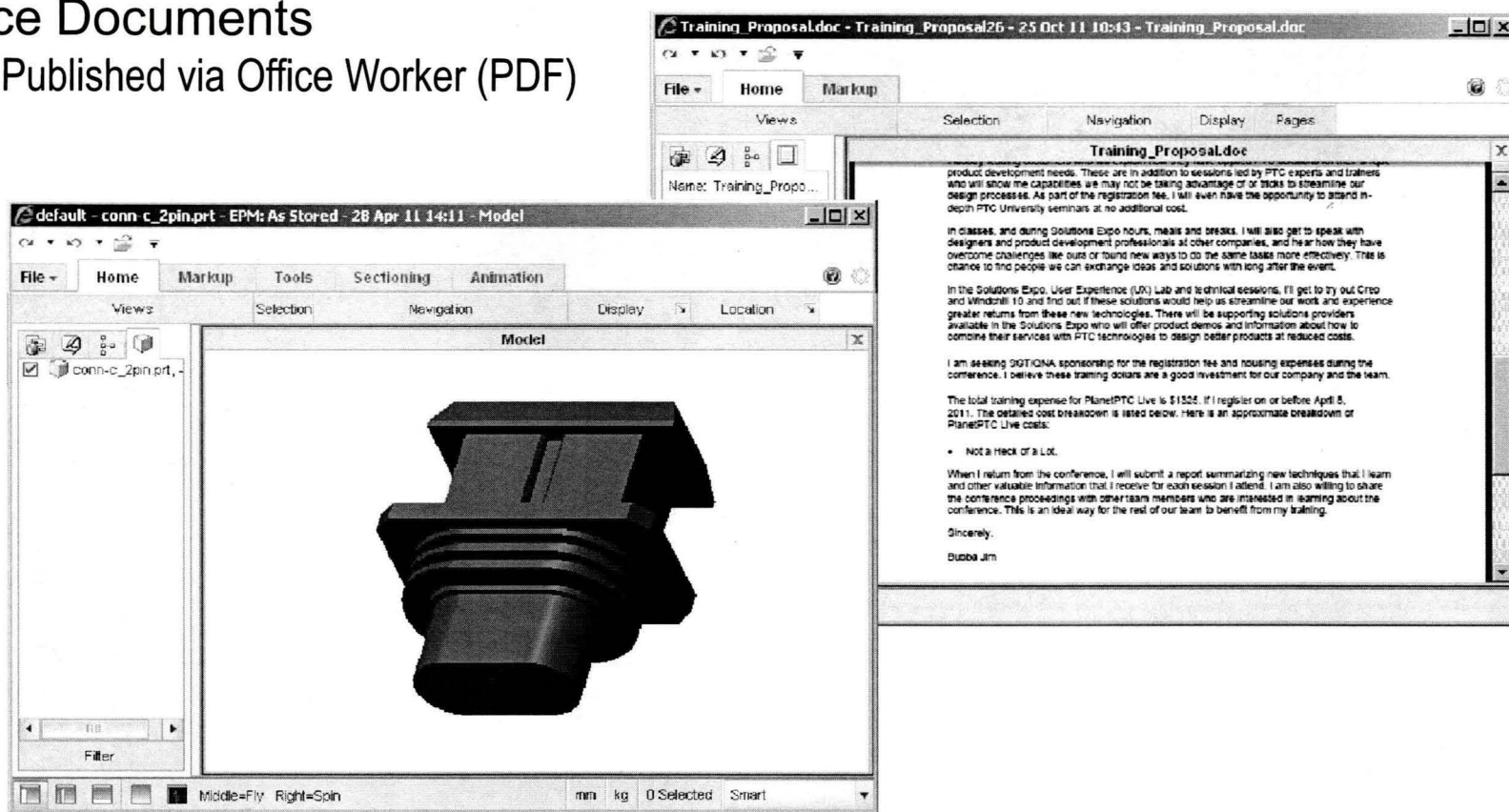
- Windchill — 9.1-M050
- ProE — WF5-M130
- Publishing Adapters — 10.1-F000
- Creo 2.0 CreoView Client — 10.1-F000
- Adobe Distiller Server 8 — 9.1-M062
- Acrobat Support Tools — 9.1-F002



CreoView Representations

Published Viz Types Discussed

- ProE Models/Assemblies/Drawings
 - Published via ProE CAD Workers (HPGL)
- Office Documents
 - Published via Office Worker (PDF)



The Issue

Is this what I think it is?

- Given a Print of a Version-Controlled Document/Drawing...
 - “Is this the **Latest Iteration** of this document?”
 - “Is it **Released**?”

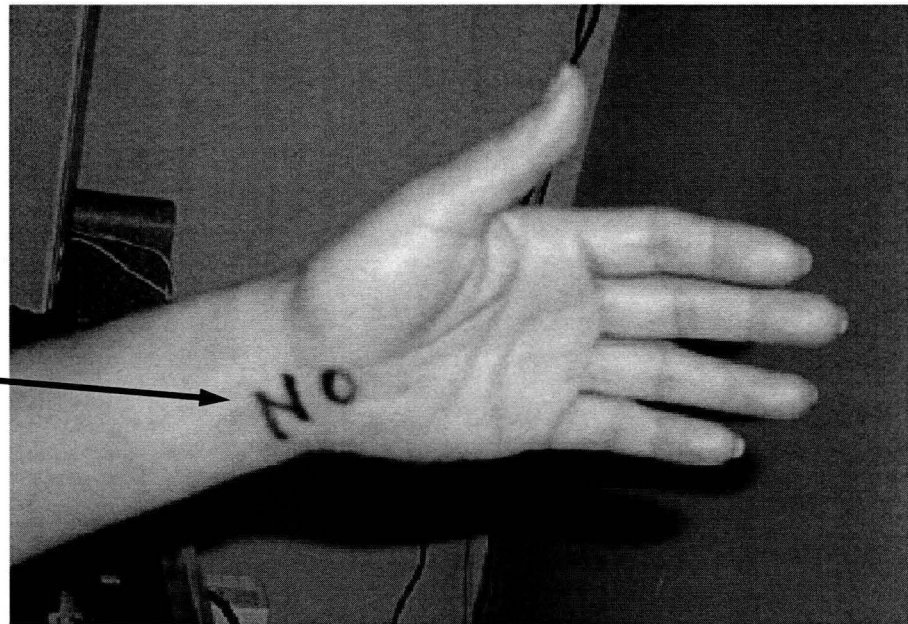


The KSC Premise

Add an Endorsement

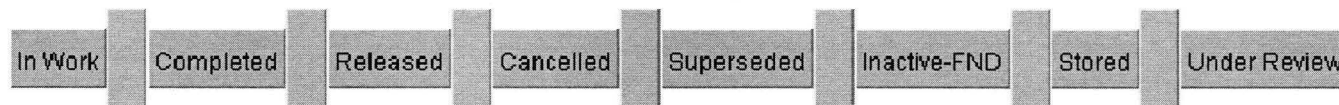
- Define an “Official” KSC Document as...
 - The **Representation** of a Windchill Document/Model/Drawing/etc.
 - Viewed/Printed via **ProductView/CreoView**
 - Containing a KSC **State-Based Watermark** to validate the document.

sample “Medical” Watermark



Defined

- A Watermark is...
 - Standard **display-time** overlay functionality in ProductView/CreoView
- For KSC, the Watermark displayed is...
 - based on the **State** of the object viewed
- The KSC standard Lifecycle States are...

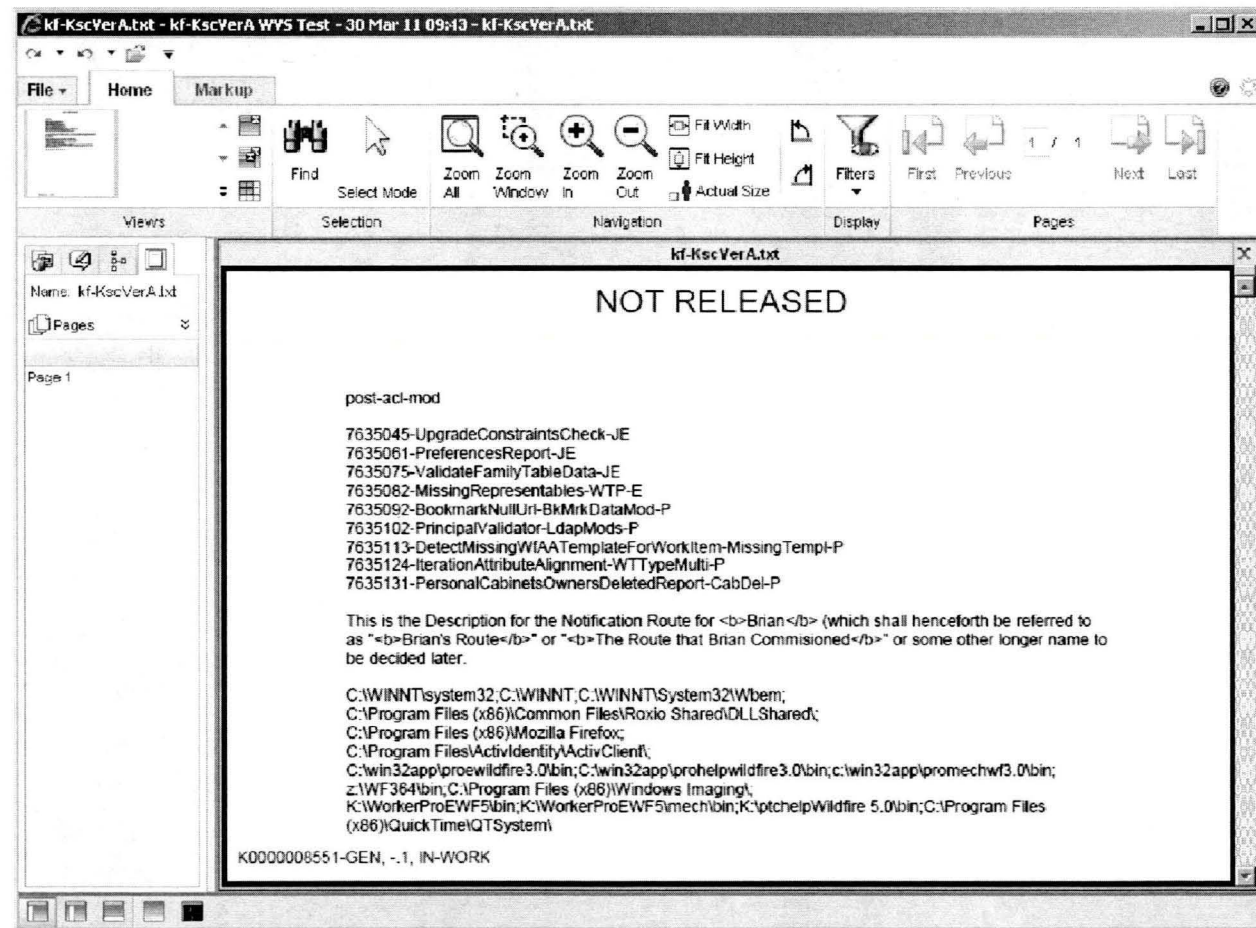


- **In-Work**, **Completed**, and **Released** are the primary working States.

KSC Watermarked Documents

WTDocuments

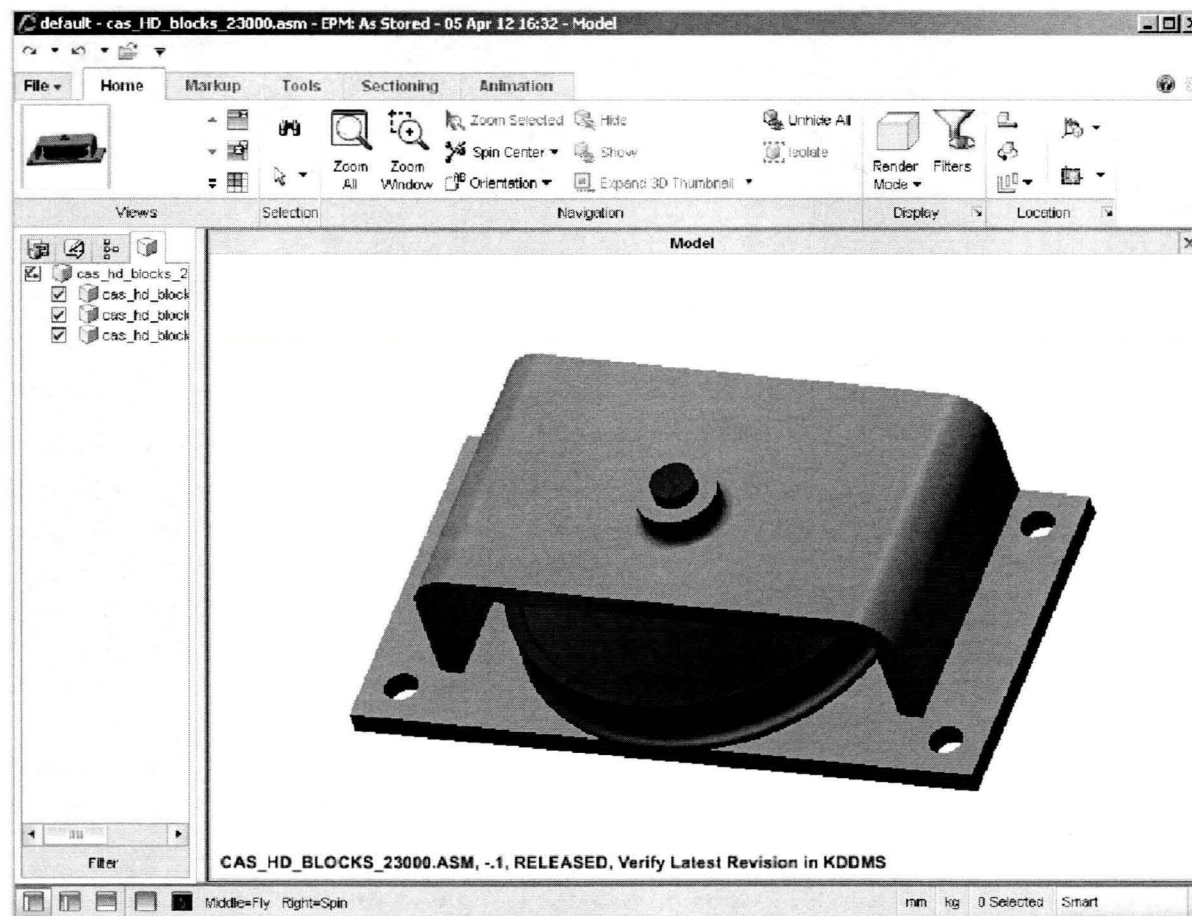
- Office Documents
 - with **State-based** Watermark
 - **IS** reliant on (PTC) **Acrobat** for Publishing
 - **IS** reliant on (PTC) **Acrobat** on **Client** for Watermarking



KSC Watermarked Documents

EPMDocuments — WTParts

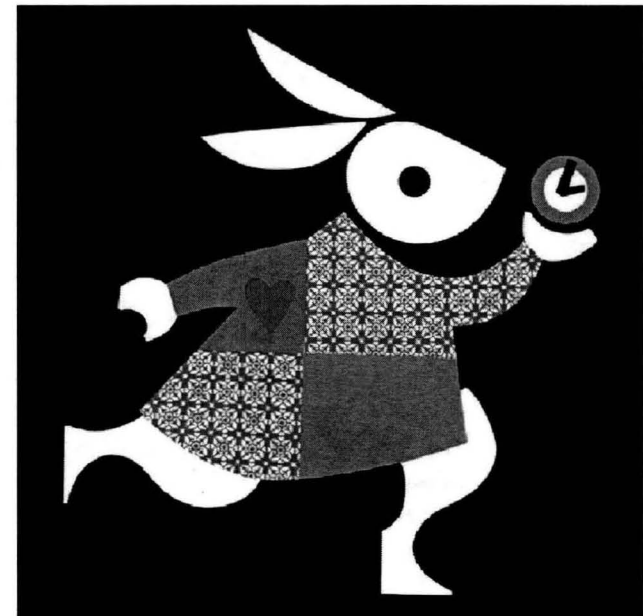
- ProE Models/Assemblies/Drawings/Structure
 - State-based Watermark — **Not** reliant on (PTC) Acrobat for Watermarking



Other Strategies

Rabbits We Have Chased

- Tried programmatically Marking “*in*” the ProE Documents
 - with Text and/or Symbols based on State
- When the Document was “Released”...
 - the in-Drawing marking did not show as such
- The dilemma of needing to iterate the Document in order to make the Released Mark appear correctly...
 - but if we did **that**, the document that was just “Released” had just gotten **Modified!**



Watermarking

What & How

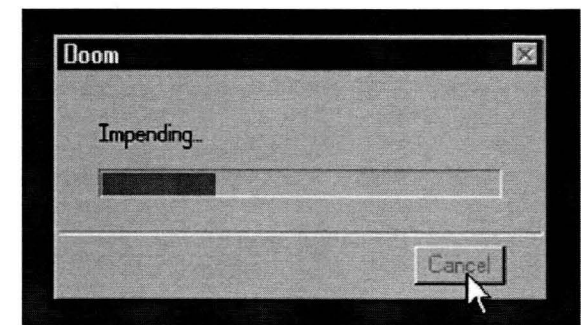
- Watermarks are Text String definitions
 - Defined to display the **String** based on the **State** of the Document

CAS_HD_BLOCKS_23000.ASM, -1, RELEASED, Verify Latest Revision in KDDMS

this Watermark

<epmdoc_number>, <epmdoc_iterationDisplayIdentifier>, RELEASED, Verify Latest Revision in KDDMS

Is based on this
Text String spec



Watermarking

What & How

- Watermarks are Text String definitions
 - Defined to display the **String** based on the **State** of the Document

CAS_HD_BLOCKS_23000.ASM, -.1, RELEASED, Verify Latest Revision in KDDMS

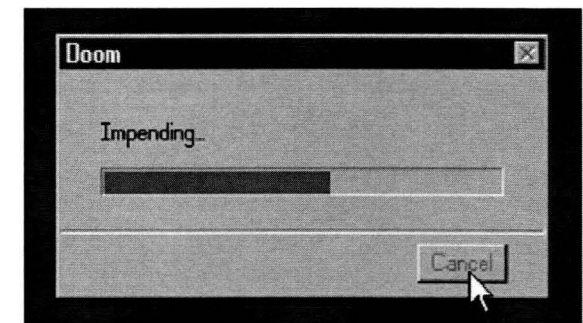
this Watermark

<epmdoc_number>, <epmdoc_iterationDisplayIdentifier>, RELEASED, Verify Latest Revision in KDDMS

- Initialization Files created for each State

- In-Work.ini
- Completed.ini
- Released.ini
- etc.

Is based on this
Text String spec



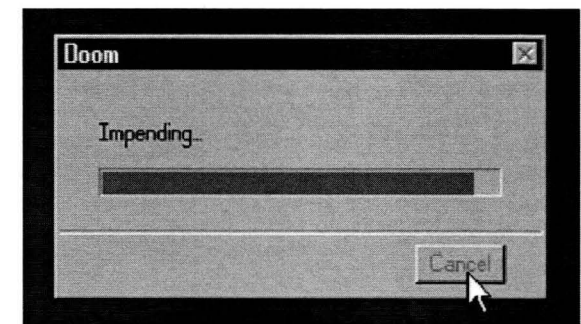
What & How

- Watermarks are Text String definitions
 - Defined to display the **String** based on the **State** of the Document

 CAS_HD_BLOCKS_23000.ASM, -1, RELEASED, Verify Latest Revision in KDDMS

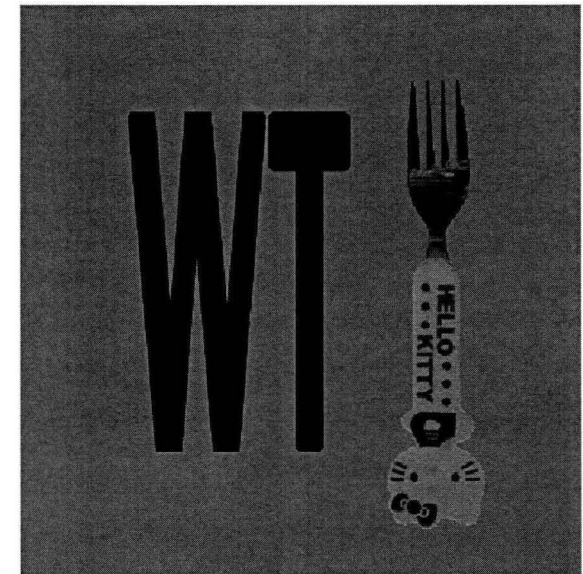
<epmdoc_number>, <epmdoc_iterationDisplayIdentifier>, RELEASED, Verify Latest Revision in KDDMS

- Initialization Files created for each State
 - In-Work.ini
 - Completed.ini
 - Released.ini
 - etc.
- Each Initialization File defines
 - **what** Text String is placed, and **where** it appears
 - built-in definitions and variable expansions are also available



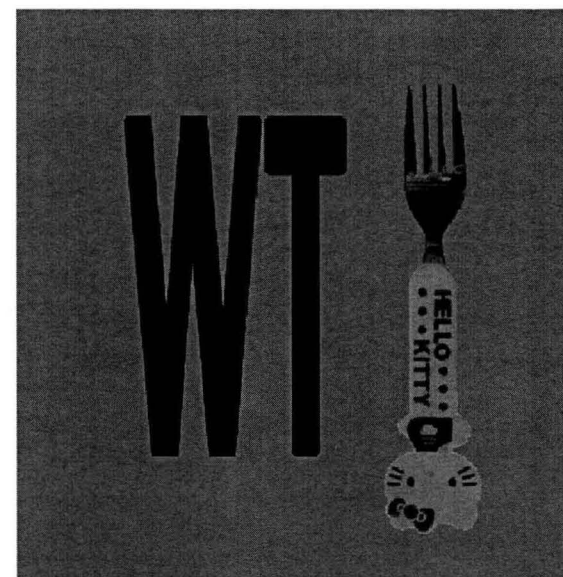
What & How — Initialization Files

- The Initialization Files are uploaded to Windchill
 - via a **ProductView Configuration**
 - Contains INI and Server Control files.
- Create a **ProductView Configuration** via the Visualization Configuration Administrator Utility
 - Creates a **collection** where the Watermark Initialization Files can be uploaded to the Windchill instance



What & How — Initialization Files

- The Initialization Files are uploaded to Windchill
 - via a **ProductView Configuration**
 - Contains INI and Server Control files.
- Create a **ProductView Configuration** via the Visualization Configuration Administrator Utility
 - Creates a **collection** where the Watermark Initialization Files can be uploaded to the Windchill instance
- Initialization Files can be uploaded as...
 - a complete collection (.zip)
 - or individually (.ini)
- Initialization Files are sent to the CreoView Client when it is opened
 - from a document's **Details Page**, for example



What & How — INI File Flavors

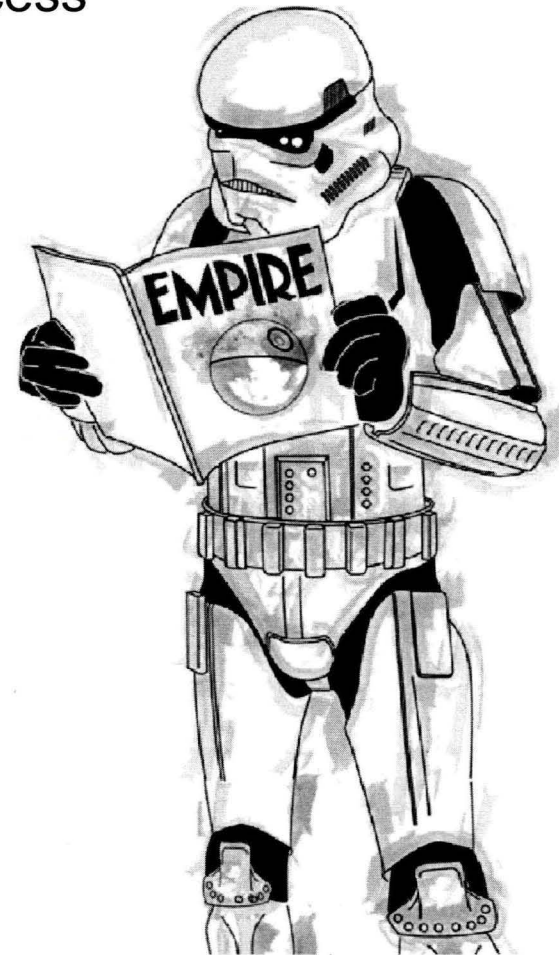
- Old School
 - Based on ProductView Standard **9.0-M110**
 - Employs **text**-based files to define Watermarks
 - Employs **text**-based Server Control files
- New School
 - Based on ProductView Lite/Pro **9.1-M062** and later
 - Employs **text**-based files to define Watermarks
 - Employs **XML**-based Server Control files
- CreoView Releases not Windchill Aligned
 - **CreoView 10.1-F000** Adapters and Clients
in use at KSC with **9.1-M050** Windchill



Getting Schooled

How — the “Official” Approach

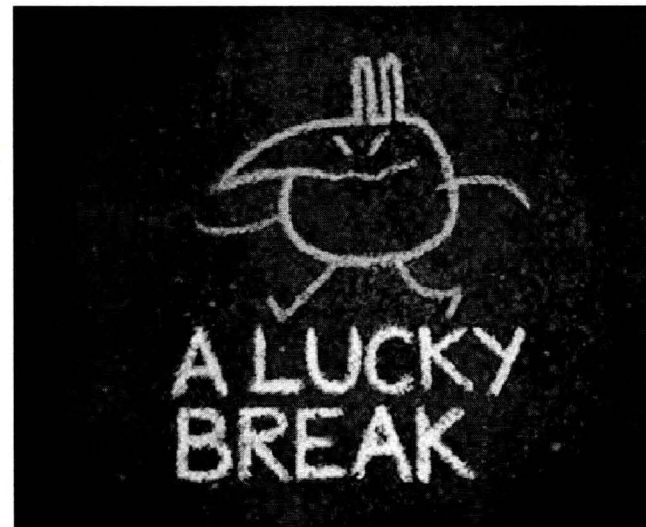
- Unraveling Watermarking is a tedious process
 - Choices are limited
 - Very little documentation
 - Examples are fuzzy
- Recommend...
 - Employ the latest Viz Builds
 - Use the latest Documentation Available
- Good Luck!



Getting Schooled

How — A **Better** Approach

- First — attend a PTC User Presentation by Alcon Labs!



Getting Schooled

How — A **Better** Approach

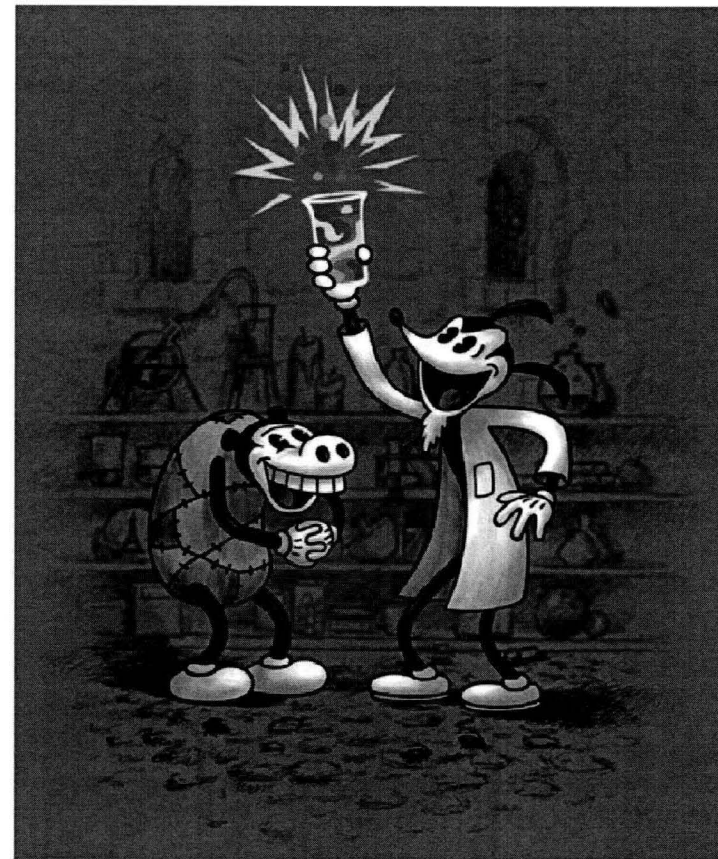
- First — attend a PTC User Presentation by Alcon Labs!
- Then — Phone a Friend!



Getting Schooled

How — A **Better** Approach

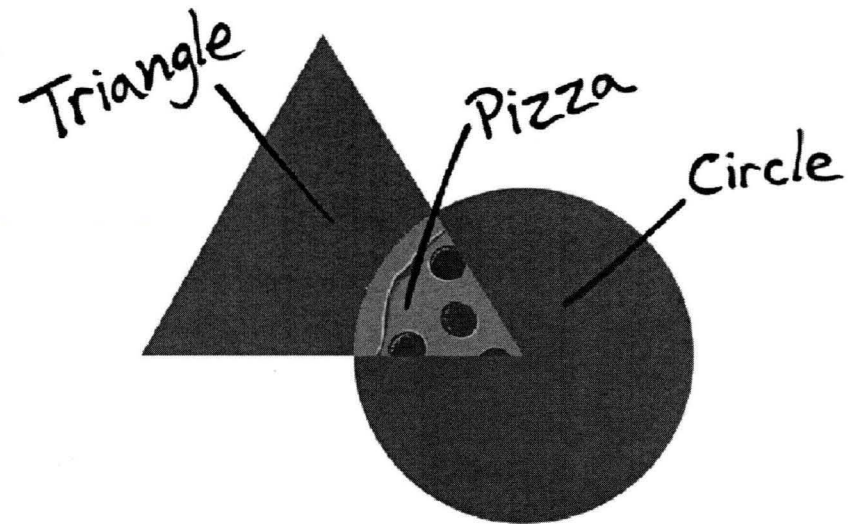
- First — attend a PTC User Presentation by Alcon Labs!
- Then — Phone a Friend!
- Learning how someone else succeeded is the **BEST** help!



Getting Schooled

How — A **Better** Approach

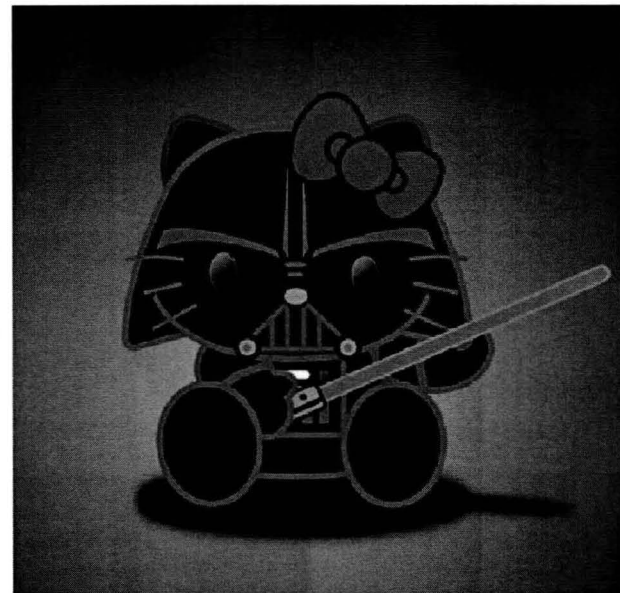
- First — attend a PTC User Presentation by Alcon Labs!
- Then — Phone a Friend!
- Learning how someone else succeeded is the **BEST** help!
- Then getting some example Data from them — **Seriously Cool!**



Creating Watermarks

Choices...

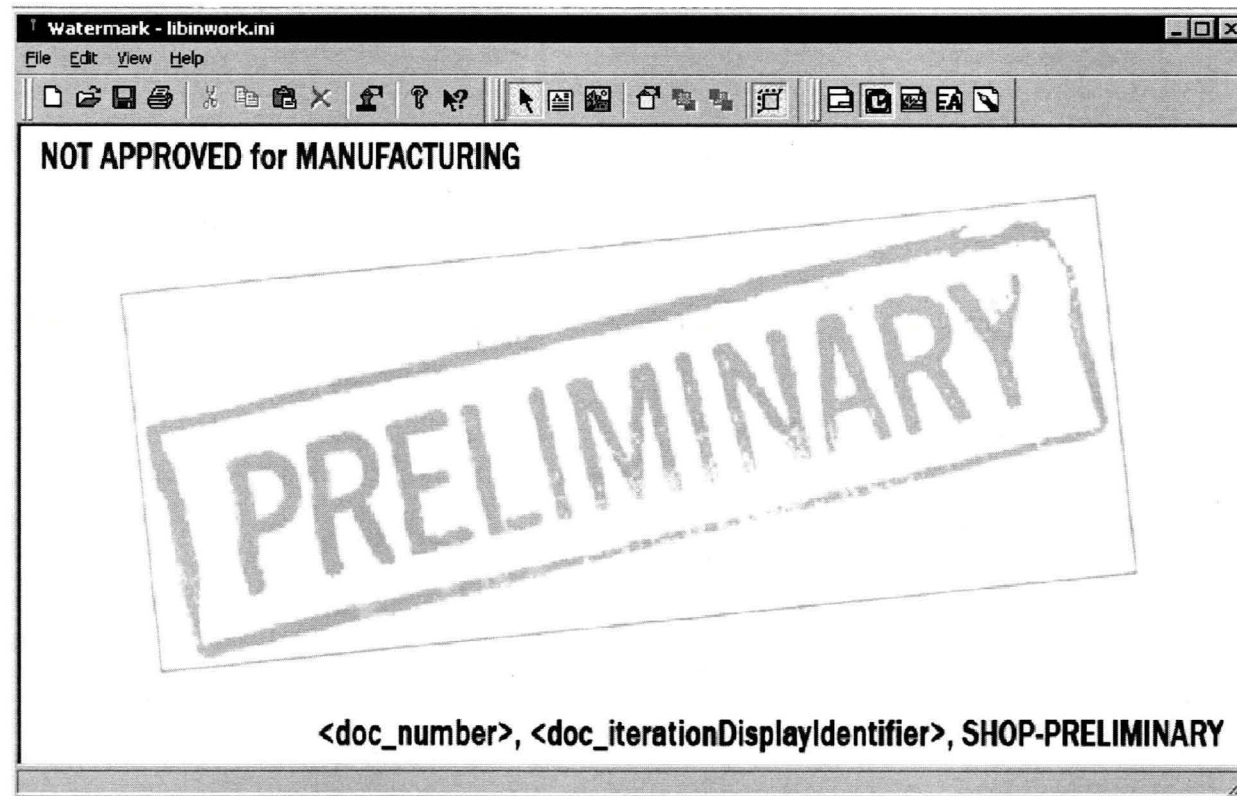
- Watermark Editor
- Manual Definition in CreoView
- Learn from Others



Creating Watermarks

Choices...

- Watermark Editor
 - for more complex layouts/features



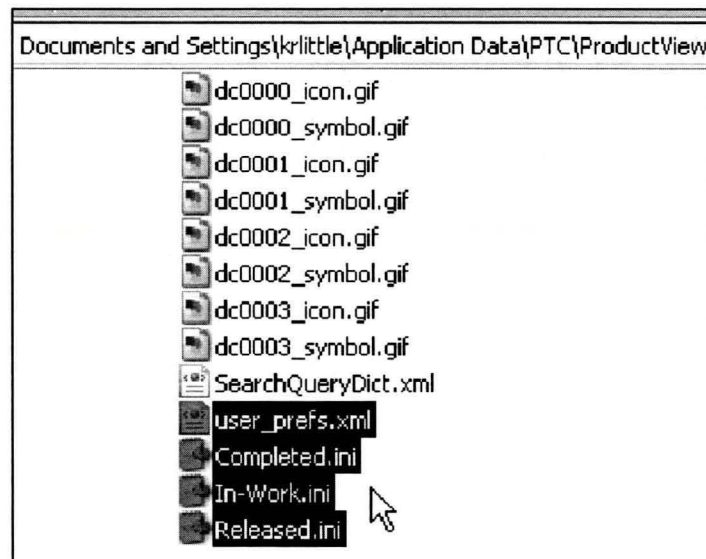
- may not be the best environment to begin in

Creating Watermarks

More Choices...

- **Manual Definition in CreoView**

- Create the Watermarks on a local workstation and Reverse-Engineer the configuration.
- Initialization/XML data saved in User's Profile.

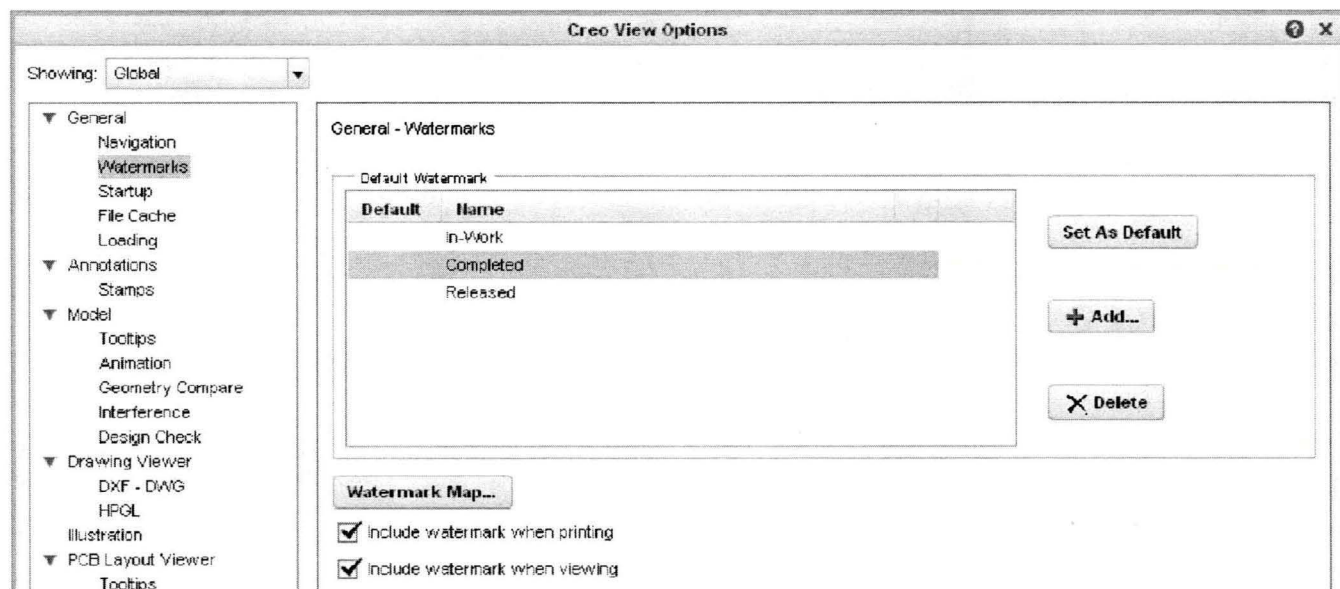


Creating Watermarks

More Choices...

■ Manual Definition in CreoView

- Create the Watermarks on a local workstation and Reverse-Engineer the configuration.
- Initialization/XML data saved in User's Profile.

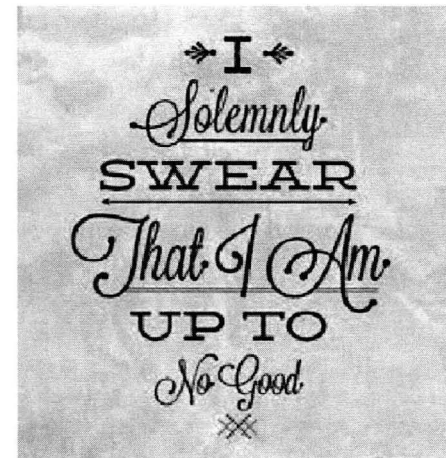


- Options saved in (local) **user_prefs.xml** — rename to **server_prefs.xml** for use in final server-based configuration.
- A valid approach for fine-tuning the data — may not be the best place to begin.

Creating Watermarks

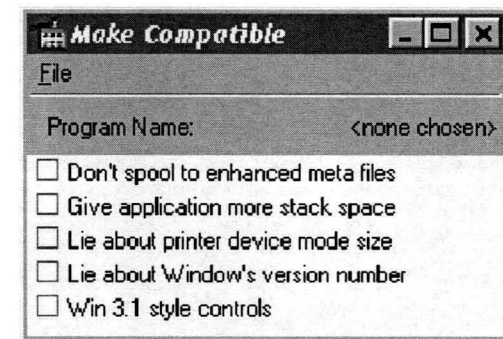
the **Preferred** Approach

- Learn from Others
 - Simply the **easiest** way to learn Watermarking
 - Start with **working** data
 - Learn the format from the INI and Server-Control files
 - Massage it based on **your** needs



INI File Format

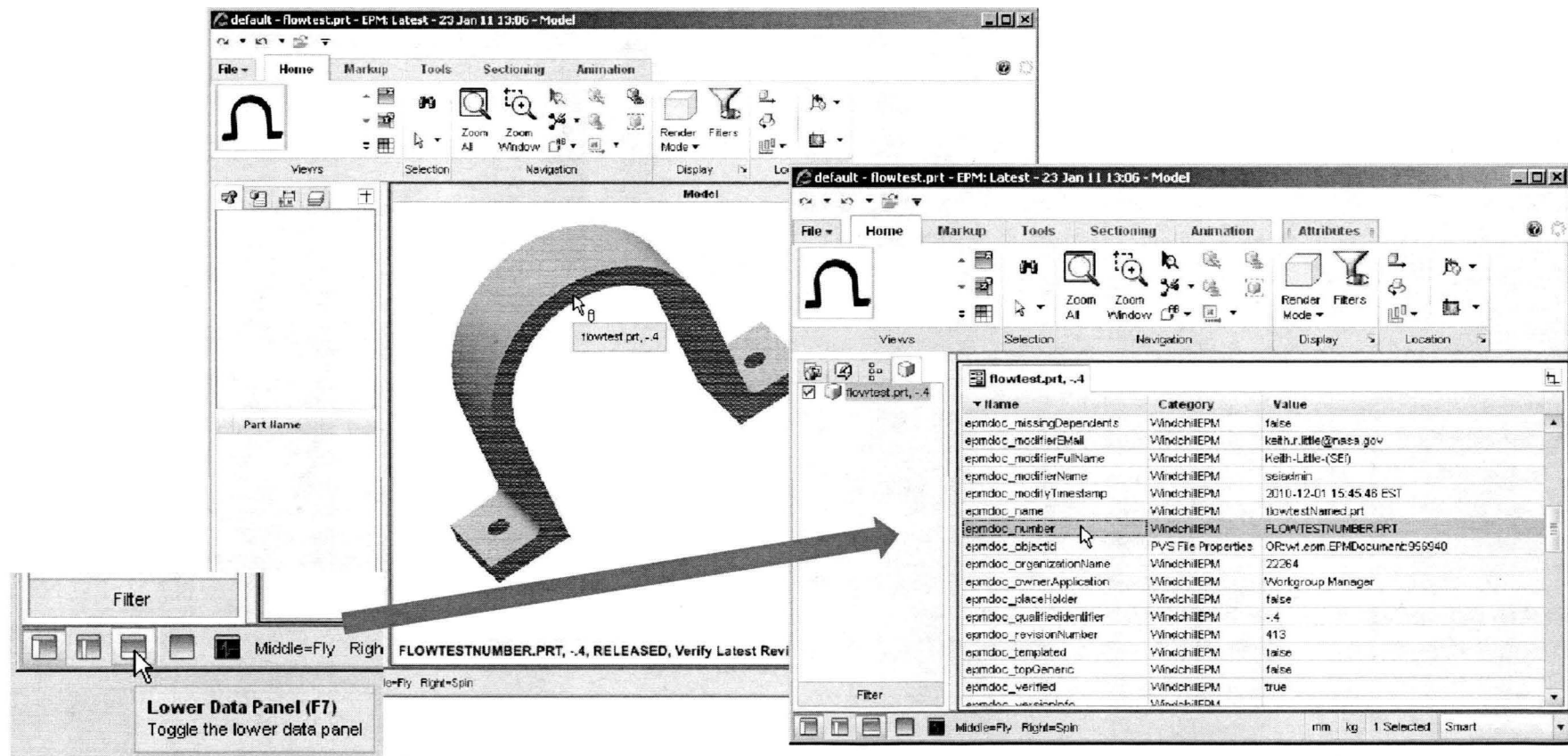
- win.ini [**Section**] format Text files
- Four Main [Doc-Type] Sections
 - [drawing], [model], [image], [document]
- Section specs for Number of Text/Images
 - [drawing]
 - **numtext=2**
- Actual Text n Specs follows for each Text String — including **<expansion>**
 - **text1**=<epmdoc_number>, <epmdoc_iterationDisplayIdentifier>, IN-WORK
 - **text2**=NOT RELEASED
- Color/Size/Font/Orientation then included
 - **color1**=255, 0, 0
 - **font1**=Arial
 - **size1**=20
 - **bold1**=true
 - ...
- Full Study by reviewing the INI files



Specification

INI File Format — knowing where to look

- Note **<epmdoc_...>** and **<doc_...>** Variable substitutions
 - e.g., text1=<epmdoc_number>, <epmdoc_iterationDisplayIdentifier>, RELEASED, Verify Latest Rev...
- Toggle the **Lower Data Panel** control — Data Chart is presented...

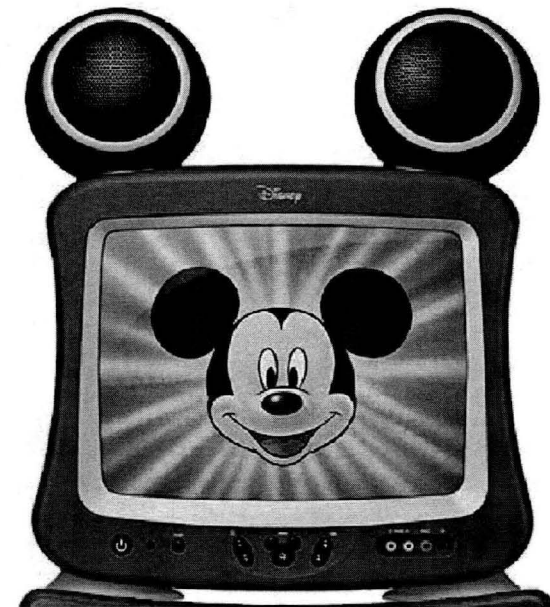


INI File Format — One Trick

- A main [Section] is being employed for both WTDocs and EPMDocs
- Use a Text String that Substitutes variables for **both** — for example...

text1=<doc_number><epmdoc_number>, <doc_iterationDisplayIdentifier><epmdoc_iterationDisplayIdentifier>, **IN-WORK**

- When it's a **WTDoc**, the <epmdoc_...> variables will have no content...
- and when it's an **EPMDoc**, the <doc_...> variables will have none.



Specification



Server Control File Format — server_prefs.xml

- XML format — contains preferences for the Client Viewer
- Four main top-level sections

```
+ <category name="GUI">  
+ <category name="Drawing Viewer">  
+ <category name="Model Viewer">  
+ <category name="General">
```

- **GUI** and **Drawing Viewer** sections have miscellaneous general data
- **Model Viewer** section contains Client Viewer size and layout options

```
- <category name="Model Viewer">  
- <preferenceList name="ViewOrientationsTable">  
- <preferenceItem>  
  <preference name="locked" value="DEFISO1" />  
  <preference name="DegZ" value="-18.3241" />  
  <preference name="DegY" value="14.0085" />  
  <preference name="DegX" value="-53.8366" />
```

- Watermark specs are in the **General** section

```
- <category name="General">  
- <category name="Watermarks">  
  <preference name="show_when_viewing" value="True" locked="true" />  
  <preference name="show_when_printing" value="True" locked="true" />  
- <preferenceList name="watermark_map">  
- <preferenceItem>  
  <preference name="watermark" value="Released" />  
  <preference name="property_value" value="Released" />  
</preferenceItem>
```

```
- <preferenceList name="watermarks">  
- <preferenceItem>  
  <preference name="file" value="Released.ini" />  
  <preference name="name" value="Released" />  
</preferenceItem>
```

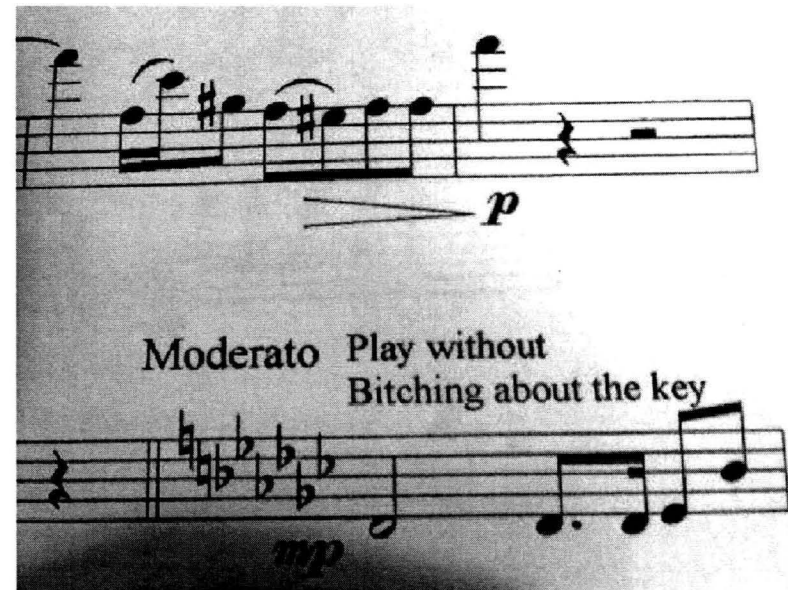
these **locked="true"** items prevent the Watermarks from being altered on the Client

Server-Side Implementation

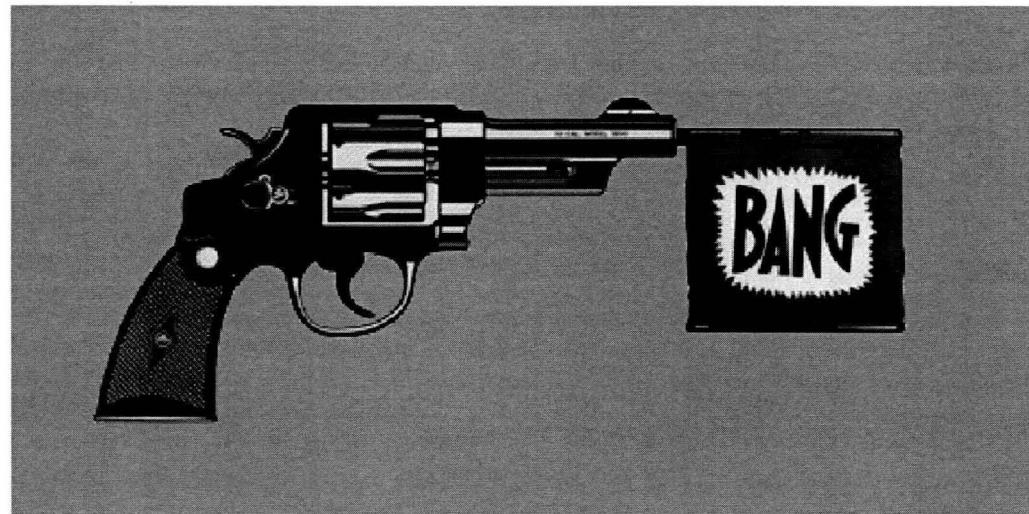
- Windchill Configuration for Watermarking
 - wvs.properties — productview.configfromserver=true



- Creation of a Watermarked Office Document
 - Office Doc is Checked-In and Published by the Office Worker (PDF)
 - User views the Office Doc via ProductView/CreoView
 - Watermark is generated as a separate PDF (based on [PTC] Client-based Acrobat Standard)
 - Published PDF plus the Watermarked PDF is displayed in ProductView/CreoView



- Creation of a Watermarked Office Document
 - Office Doc is Checked-In and Published by the Office Worker (PDF)
 - User views the Office Doc via ProductView/CreoView
 - Watermark is generated as a separate PDF (based on [PTC] Client-based Acrobat Standard)
 - Published PDF plus the Watermarked PDF is displayed in ProductView/CreoView
- Means ProductView/CreoView **and** the Acrobat Tools have to be installed on **each** Client for Office Document Watermarks to display correctly.
- 10.1 = Adobe LiveCycle



- Watermarks are a reasonably simple way to add reliability to your Documents.
 - Start Simple
 - Don't Re-Invent the Wheel
 - Learn from Others

